Integration of Advanced Life Support Control Systems

David Overland
NASA Johnson Space Center
Dr. Karlene Hoo
Texas Tech University
Marvin Ciskowski
Hamilton Standard

Integrated Testing

- The development of stand-alone systems is relatively well understood
- When you put systems together, interesting and unpredictable things happen
  - Specifically, the integrated system behaves differently than the stand-alone systems behave separately
- When we have done this before, actually integrating these systems has driven out requirements and identified technology gaps in the ALS program
- Based on that experience, we have identified control system architecture and integration as a critical technology gap
- AIM proposed an integrated test to explore the design constraints and integration requirements of control systems

Test Objectives

- AIM Test was designed
  - To stress interfaces
  - To identify information flows
  - To explore operations concepts and dependencies
  - To investigate architecture capabilities and requirements
- Intent is also to determine what types of data and autonomous capabilities will be required by crew, vehicle and ground control during complex mission scenarios
  - What decisions must be made, where are they made, what information is needed to make those decisions, how does the information get there, and how reliable is the information
  - These need to be determined to identify whether infrastructure and architecture can support such capabilities

Test Components

- Controls Investigation
  - WRS preprocessor systems with independent control systems for each reactor
    - Aerobic bioreactor
    - Anoxic bioreactor
- ARS simulation
- Scenario development and Task analysis
  - Mapping command and data flows to capabilities
- Narrative Integration

Controls Investigation

- The two bioreactors "stand in" for any two interdependent systems
  - e.g. ARS and WRS
  - Flight systems are developed independently
    - Separate System Requirements Specifications
    - Interfaces are defined and controlled
    - Separate subcontractor organizations
    - Developed at different times in the program
- Question:
  - What requirements must be levied on each system to enable integration of the control systems?

Controls Development Process

- Three prerequisites are required to develop a process control system
  - Process must be steady-state stable
  - Process must be controllable
    - i.e. there must be control (dependent) parameters and manipulated (independent) parameters
  - Process must be observable
    - i.e. there must be observable parameters that correspond to the controlled parameters

Controls Models

- There are also three types of models/analysis required for process control development
  - Stochiometric model
  - Equilibrium model
  - Control-relevant model
    - Dependent on optimization criteria
- These models are *necessary* to design a controllable system

Lessons Learned from Integrated Testing

Lesson 1 is that systems must be *designed* for controllability

- *Control* in this context means bringing the process back into equilibrium in the desired optimization range when the process is perturbed by input or environmental variations
- Controllability is design sensitive
  - Therefore the control design precedes the hardware design of the system
  - Controllability and observability dictate the sensor selection and placement
- This is in contrast to spacecraft avionics design, where software requirements are derived *from* the hardware design

Control (A) + Control (B) ≠ Control (A+B)

Lesson 2 is that

- Controllability is not additive for interdependent systems
  - Analysis and Design must encompass entire system
  - Analysis of system components provides no information about system controllability

Interdependence Causes Complexity

Lesson 3 is that the complexity of integrating the bioreactor control systems is not just an attribute of the biology, but also of the interdependence of the processes

- *Process* in this context refers to a transformation of something to something else
  - Processes have rates, control variables and dependent variables
- *Interdependence* means that changes in the parameters of one system necessitate changes in the controls of another system, either
  - automatically (as in the case of the bioreactors) or
  - by intent (a manual or autonomous command)

Conclusion

- These lessons can be true of autonomous and automated systems
- The possibility of instability is one of the drivers to disallow automation of on-board systems
  - System dependencies are often discovered *in-situ*, after deployment
  - Automation can enable those dependencies
  - Autonomy and automation added after subsystem design can also generate dependencies between subsystems that were designed to be independent

Importance

- The Constellation Program has autonomy and automation requirements different from previous programs
- The *design* of such systems requires risk mitigation and engineering strategies different from previous programs
  - This is not the same as developing autonomy and automation technology
- Subsystem requirements must be derived from integrated design
  - Development of subsystem *specifications* in contrast to subsystem functional *requirements* and constraints